ABSTRACT

An efficient write-watch mechanism and process. A bitmap is associated with the virtual address descriptor (VAD) for a process, one bit for each virtual page address allocated to a

5 process having write-watch enabled. As part of the write-watch mechanism, if a virtual address is trimmed to disk and that virtual address page is marked as modified, then the corresponding bit in the VAD is set for that virtual address page. In response to an API call (e.g., from a garbage

10 collection mechanism) seeking to know which virtual addresses in a process have been modified since last checked, the memory manager walks the bitmap in the relevant VAD for the specified virtual address range for the requested process. If a bit is set, then the page corresponding to that bit is known to have

15 been modified since last asked. If specified by the API, the bit is cleared in the VAD bitmap so that it will reflect the state since this time of asking. If the bit is not set, to determine if the page was modified, the page table entry (PTE) is checked for that page, and if the PTE indicates the page

20 was modified, the page is known to be modified, otherwise that page is known to be unmodified since the last call. One enhancement uses page directory tables to locate a series of trimmed pages, sometimes avoiding the need to access the PTE.